

**What is Claimed:**

1. A virtualization system for a host computer having at least one host processor and system resources including physical I/O hardware and memory divided into most privileged system memory and less privileged user memory, the system comprising:

virtualization software that operates in said less privileged user memory and divides said host computer into a plurality of virtual partitions including at least one user guest partition that provides a virtualization environment for at least one guest operating system and at least one input/output (I/O) partition that maps said physical I/O hardware to endpoints of an I/O channel server in said at least one I/O partition, said I/O channel server sharing the physical I/O hardware with at least one guest partition via a memory channel comprising memory shared between said at least one I/O partition and said at least one guest partition;

a resource database for use in managing use of said at least one host processor and said system resources;

at least one monitor that operates in said most privileged system memory and maintains guest applications in said at least one guest partition within memory space specified in said resource database; and

a context switch between said at least one monitor and said respective guest and I/O partitions for controlling multitask processing of software in said partitions on said at least one host processor.

2. The virtualization system of claim 1, wherein upon receipt of a request to said I/O channel server from said at least one guest partition to access physical I/O hardware said I/O partition checks with partition descriptors stored in a monitor associated with said at least one guest partition to verify that the requested physical I/O hardware access is valid.

3. The virtualization system of claim 1, wherein said mapping by said at least one I/O partition of said physical I/O hardware of said host computer to endpoints of said I/O channel server in said I/O partition is performed by I/O partition software that multiplexes through shared common I/O physical hardware any I/O requests to said common I/O physical hardware from multiple partitions connected to said I/O partition by respective memory channels.

4. The virtualization system of claim 3, wherein an I/O monitor associated with said I/O partition implements a system call interface between said I/O monitor and said I/O partition, said system call interface converting and validating client partition relative addresses, obtained as

buffer parameters of requests sent through respective memory channels from client memory channel drivers, as valid hardware physical addresses of memory currently assigned to the client partition requesting access to said common I/O physical hardware.

5. The virtualization system of claim 4, wherein messages between a server of said I/O partition and said respective guest partitions are queued by the guest partitions and de-queued by the I/O partition server and the partition relative physical addresses are converted by the I/O partition server to physical I/O hardware addresses with the aid of the I/O monitor, whereby data may be exchanged with hardware I/O adapters connected between said I/O monitor and said common I/O physical hardware.

6. The virtualization system of claim 1, wherein said mapping by said at least one I/O partition of said physical I/O hardware of said host computer to endpoints of said I/O channel server in said I/O partition is performed by passing I/O setup information via said memory channel to said I/O channel server so as to set up a high performance memory channel between a client partition requesting I/O access and intelligent physical I/O hardware and sending data directly between said client partition requesting I/O access and said intelligent physical I/O hardware via said high performance memory channel.

7. The virtualization system of claim 6, wherein the guest partition requesting I/O access transfers data via said I/O memory channel to said intelligent physical I/O hardware using one of a user mode I/O or direct memory access data transfer operation.

8. The virtualization system of claim 1, wherein the at least one I/O partition includes two redundant I/O partitions.

9. A method of managing I/O operations of a plurality of operating system instances on a host computer having at least one host processor and system resources including physical I/O hardware, the method comprising the steps of:

dividing said host computer into a plurality of virtual partitions including at least one user guest partition that provides a virtualization environment for at least one guest operating system and at least one input/output (I/O) partition that maps said physical I/O hardware to endpoints of an I/O channel server in said at least one I/O partition;

said I/O channel server sharing the physical I/O hardware with at least one guest partition via a memory channel comprising memory shared between said at least one I/O partition and

said at least one guest partition;

tracking allocation of said memory channel in a resource database.

10. The method of claim 9, further comprising the step of checking with partition descriptors stored in a monitor associated with said at least one guest partition to verify that a requested physical I/O hardware access is valid.

11. The method of claim 9, wherein said mapping by said at least one I/O partition of said physical I/O hardware of said host computer to endpoints of said I/O channel server in said I/O partition includes the steps of multiplexing through shared common I/O physical hardware any I/O requests to said common I/O physical hardware from multiple partitions connected to said at least one I/O partition by respective memory channels.

12. The method of claim 11, further comprising the steps of implementing a system call interface between an I/O monitor and said I/O partition and said system call interface converting and validating client partition relative addresses, obtained as buffer parameters of requests sent through respective memory channels from client memory channel drivers, as valid hardware physical addresses of memory currently assigned to the client partition requesting access to said common I/O physical hardware.

13. The method of claim 12, further comprising the steps of the guest partitions queuing messages between a server of said I/O partition and said respective guest partitions, the I/O partition server de-queuing the partition relative physical addresses are converting the partition relative physical addresses to physical I/O hardware addresses, and exchanging data with hardware I/O adapters connected between said I/O monitor and said common I/O physical hardware.

14. The method of claim 9, wherein the step of mapping by said at least one I/O partition of said physical I/O hardware of said host computer to endpoints of said I/O channel server in said I/O partition includes the steps of passing I/O setup information via said memory channel to said I/O channel server so as to set up a high performance memory channel between a client partition requesting I/O access and intelligent physical I/O hardware and sending data directly between said client partition requesting I/O access and said intelligent physical I/O hardware via said high performance memory channel.

15. The method of claim 14, wherein the step of sending data directly between said client partition and said intelligent physical I/O hardware comprises the step of using one of a user mode I/O and a direct memory access data transfer operation.